AMENDMENTS TO THE CLAIMS

Claim 1 (currently amended): An apparatus for coordinating communications between a plurality of tightly coupled processors, comprising:

one or more generalized queues communicatively couplable to the tightly coupled processors for storing entries to be passed between the tightly coupled processors, each queue having a start location, an end location, a put pointer for indicating a next location in the queue into which an entry is to be supplied, and a get pointer for indicating a next location in the queue from which a entry is be received; and

one or more specialized registers communicatively couplable to the tightly coupled processors for assisting the tightly coupled processors in utilizing the generalized queues for interprocessor communications and adapting the generalized queues to match a current operating environment;

wherein the generalized queues and specialized registers are content-configurable in accordance with programs executable in the tightly coupled processors.

Claim 2 (original): The apparatus as recited in claim 1, further comprising at least one specialized register for storing the put pointer and the get pointer of a generalized queue associated with the specialized register to assist in supplying entries into the generalized queue, receiving entries from the generalized queue, and determining whether the generalized queue is empty, full, not full, or not empty.

Claim 3 (original): The apparatus as recited in claim 1, further comprising at least one specialized register for storing an end location of a generalized queue associated with that specialized register to dynamically adapt a size of the generalized queue to the current operating environment.

Claim 4 (original): The apparatus as recited in claim 1, further comprising at least one specialized register for storing requests from two or more processors to reset a generalized queue being utilized to pass entries between the processors, in order to facilitate a coordinated reset of that generalized queue.

Claim 5 (original): The apparatus as recited in claim 1, further comprising at least one specialized register for storing attention conditions from one processor destined for another processor.

Claim 6 (original): The apparatus as recited in claim 1, further comprising at least one specialized register for storing whether the generalized queues are not empty or not full.

Claim 7 (original): The apparatus as recited in claim 4, further comprising at least one specialized register for storing an enable indicating whether an attention condition destined for one processor will be visible to that processor.

Claim 8 (original): The apparatus as recited in claim 4, further comprising at least one specialized register for storing an enable indicating whether an indicator of whether the generalized queue is not empty or not full destined for one processor will be visible to that processor.

Claim 9 (currently amended): ~~An interface controller chip comprising the~~The apparatus of claim 1, wherein the apparatus is housed in an interface controller chip, and wherein the one or more tightly coupled processors are for providing I/O processing and physical connectivity between a host device coupled to a host bus and external data storage devices coupled to one or more storage area networks.

Claim 10 (currently amended): [[A]]The apparatus of claim 9, wherein the interface controller chip is housed in a host bus adapter (HBA) ~~comprising the interface controller chip of claim 9~~, and wherein the host bus is a PCI or PCI-X bus and the external data storage devices communicate over the one or more storage area networks using fibre channel (FC) protocols.

Claim 11 (currently amended): [[A]]The apparatus of claim 10, wherein the HBA is housed in a server computer ~~comprising the HBA of claim 10~~.

Claim 12 (original): In a multi-processor system including a plurality of tightly coupled processors and one or more generalized queues communicatively couplable to the tightly coupled processors for storing entries to be passed between the tightly coupled processors, an apparatus for coordinating communications between the tightly coupled processors, comprising:

one or more specialized registers communicatively couplable to the tightly coupled processors and configurable in accordance with programs executable in the tightly coupled processors for

adjusting a size and location of the generalized queues to dynamically adapt the generalized queue to the current operating environment;

informing the tightly coupled processors when the generalized queues are empty, full, not empty, or not full; and

enabling attention conditions to be passed between the tightly coupled processors.

Claim 13 (currently amended): An apparatus for providing I/O processing and physical connectivity between a host device coupled to a host bus and external data storage devices coupled to one or more storage area networks, comprising:

a plurality of tightly coupled processors for coordinating a transfer of information between the host device and the external storage devices;

one or more generalized queues communicatively couplable to the tightly coupled processors for storing entries to be passed between the tightly coupled processors; and

one or more specialized registers communicatively couplable to the tightly coupled processors and configurable in accordance with programs executable in the tightly coupled processors for assisting the tightly coupled processors in utilizing the generalized queues for interprocessor communications and adapting the generalized queues to match a current operating environment;

wherein the generalized queues and specialized registers are content-configurable in accordance with programs executable in the tightly coupled processors.

Claim 14 (original): The apparatus as recited in claim 13, wherein the tightly coupled processors are programmed for utilizing the one or more specialized registers to:

adjust a size and location of the generalized queues to dynamically adapt the generalized queue to the current operating environment;

inform the tightly coupled processors when the generalized queues are empty, full, not empty, or not full; and

enable attention conditions to be passed between the tightly coupled processors.

Claim 15 (currently amended): A method for coordinating communications between a plurality of tightly coupled processors, comprising:

configuring one or more generalized queues to hold a particular type of content;

configuring one or more specialized registers to assist the tightly coupled processors in adapting and utilizing the one or more generalized queues for storing and passing entries between the tightly coupled processors and facilitating interprocessor communications;

storing and retrieving information in the configured specialized registers for use in adapting the generalized queues to match a current operating environment;

storing and retrieving information in the configured specialized registers for use in determining when entries may be supplied into or received from the generalized queues; and

storing and retrieving attention conditions in the configured specialized registers to be passed between the tightly coupled processors.

Claim 16 (original): The method as recited in claim 15, wherein for each generalized queue, the method further comprises storing, in the configured specialized registers, a put pointer for indicating a next location in the queue into which an entry is to be supplied and a get pointer for indicating a next location in the queue from which a entry is be received, the get and put pointers for assisting in supplying entries into the generalized queue, receiving entries from the generalized queue, and determining whether the generalized queue is empty, full, not full, or not empty.

Claim 17 (original): The method as recited in claim 15, wherein for each generalized queue, the method further comprises storing, in the configured specialized registers, an end location of the generalized queue for dynamically adapting a size of the generalized queue to the current operating environment.

Claim 18 (original): The method as recited in claim 17, wherein for each generalized queue, the method further comprises storing, in the configured specialized registers, requests from two or more processors to reset the generalized queue in order to facilitate a coordinated reset of that generalized queue.

Claim 19 (original): The method as recited in claim 18, further comprising changing the size of a generalized queue by performing a coordinated reset of the generalized queue and storing a new end location in the configured specialized registers.

Claim 20 (original): The method as recited in claim 15, further comprising storing, in the configured specialized registers, attention conditions from one processor destined for another processor.

Claim 21 (original): The method as recited in claim 15, further comprising storing, in the configured specialized registers, whether the generalized queues are not empty or not full.

Claim 22 (original): The method as recited in claim 15, further comprising storing, in the configured specialized registers, an enable indicating whether an attention condition destined for one processor will be visible to that processor.

Claim 23 (original): The method as recited in claim 15, further comprising storing, in the configured specialized register, an enable indicating whether an indicator of whether the generalized queue is not empty or not full destined for one processor will be visible to that processor.

Claim 24 (original): In a multi-processor system including a plurality of tightly coupled processors and one or more generalized queues communicatively couplable to the tightly coupled processors for storing entries to be passed between the tightly coupled processors, a method for coordinating communications between the tightly coupled processors, comprising:

configuring one or more specialized registers to assist the tightly coupled processors in adapting and utilizing the generalized queues for storing and passing entries between the tightly coupled processors and facilitating interprocessor communications;

adjusting a size and location of the generalized queues in accordance with information stored in the specialized registers to dynamically adapt the generalized queues to the current operating environment;

informing the tightly coupled processors when the generalized queues are empty, full, not empty, or not full in accordance with information stored in the specialized registers; and

passing attention conditions between the tightly coupled processors in accordance with information stored in the specialized registers.